

INTEGRATING APP-V WITH MICROSOFT® VDI



©2013 Microsoft Corporation. All rights reserved. This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Some examples are for illustration only and are fictitious. No real association is intended or inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

Table of Contents

Introduction.....	4
VDI (Session-Based) with App-V	4
VDI (VM-Based) with App-V	4
Desktop Virtualization Components.....	6
Microsoft VDI.....	6
Microsoft App-V	6
Configuration Manager	6
User State Virtualization	6
Benefits of App-V and Microsoft VDI.....	8
VDI (Session based) Benefits	8
VDI (VM based) Benefits.....	9
Microsoft VDI with App-V Architecture.....	10
App-V Delivery Infrastructure.....	11
VDI (Session and VM-based) Collections	12
Client Access Infrastructure.....	12
Common Components and Configurations.....	13
RemoteApp and Desktop Connections	14
Integrating Microsoft VDI with App-V 5.0.....	16
Integrating App-V with RDSH	17
Deploying the RDSH Role.....	18
Deploying the App-V RDS Client	18
Virtual Application Deployment to Remote Desktop Servers.....	19
Reduce RDSH Storage Requirements	20
Publishing App-V apps with RemoteApp.....	22
Integrating App-V in Personal VDI (VM based) Collections	24
Deploying the App-V Client.....	24
Virtual application Deployment to Personal VDI (VM based).....	26
Reduce Personal VDI (VM based) Storage Requirements	26
Alternate Application Delivery to VDI (VM based).....	28
Integrating App-V in Pooled Virtual Machine Collections	28
Deploying the App-V Client.....	29
Virtual Application Publishing to Pooled VDI (VM-based).....	30
Reduce Pooled VDI (VM-based) Storage Requirements.....	31
Alternate Application Delivery to VDI (VM-based)	33
Conclusion	34
Appendix	35
Deploying the App-V Client via Configuration Manager.....	35
Integrating App-V with Microsoft VDI	2

Deploying the App-V Client Manually	39
Additional App-V Client Configuration Options	41

Introduction

Enterprise organizations are facing the growing presence of consumer devices in the workplace. Both session- and VM-based Virtual Desktop Infrastructure (VDI) technologies have matured, enabling IT to deliver comprehensive solutions that support the unique requirements of consumer devices. Microsoft® VDI technologies combined with App-V 5.0, SCCM 2012 SP1 and user-state virtualization provide the solutions necessary to support today's modern work styles.

Implementing App-V with Microsoft VDI provides benefits for organizations that either currently use VDI, or are considering VDI implementation. As organizations move towards VDI, there are new challenges that must be resolved to provide a scalable and supportable solution.

Combining App-V with VDI enables administrators to reduce the number of servers required for VDI (session-based), the number of virtual desktop images for their VDI (VM-based) and the overall storage requirements across both solutions.

Microsoft Application Virtualization 5.0 allows organizations to realize the benefits of App-V in Microsoft VDI solutions (session-based and VM-based) and take advantage of features specifically designed for VDI (Shared Content Store mode). This document provides an overall explanation of Desktop Virtualization as well as specific instruction on integration of App-V 5.0 with VDI (session-based) implemented on Windows® Server 2012 Microsoft Remote Desktop Session Host (RDSH) and VDI (VM-based) implemented on Remote Desktop Virtualization Host (RDVH) virtual desktops. The following sections provide an overview of the impact of App-V with Microsoft VDI.

VDI (Session-Based) with App-V

VDI (session-based) includes shared desktops and RemoteApp solutions, which enable delivery of applications to users from a Remote Desktop Protocol (RDP) client. Planning and implementing VDI (session-based) with Microsoft RDSH requires identifying the target applications, testing the applications, and remediating any conflicts. With VDI (session-based) without App-V, organizations must create server silos to host incompatible applications that do not work with other targeted applications. Microsoft App-V for Remote Desktop Services isolates, reduces, and in some cases eliminates, silos allowing customers to increase user density and reduce TCO in VDI (session-based) solutions, while enabling certain incompatible software.

VDI (VM-Based) with App-V

VDI (VM-based) including pooled (stateless) and personal (stateful) VDI (VM-based) enable administrators to provide an entire desktop computing environment from a centralized datacenter to any location. Personal VDI (VM-based) operate like a physical desktop in that changes (application installation, user data, and settings) are persisted. However, providing everyone in an organization access to a personal desktop is expensive since each personal VDI

(VM-based) stores copies of applications in each VM in the central datacenter, using expensive storage.

Because personal virtual desktops are cost prohibitive, companies may implement pooled VDI (VM-based), where VMs are not assigned to specific users and can be reused by any users with access to the pooled collection. With user-state virtualization, user changes (settings and data) can be persisted in VDI (VM based); however, application installations are not persisted.

App-V 5.0 provides benefits including isolation and virtualization that reduce the testing and storage requirements for applications presented into VDI (VM-based), and simplify application delivery with on-demand application streaming and no local caching with shared content store mode.

Desktop Virtualization Components

Building complete desktop virtualization solutions including Microsoft VDI and App-V requires a combination of technologies. This guide focuses on specific solutions and instruction for integrating App-V with the respective technology. Below is a list of the desktop virtualization solution components and links to additional information.

Microsoft VDI

Microsoft VDI allows users seamless access to their rich and full fidelity Windows environment running in the datacenter, from any device. Microsoft VDI is powered by Windows Server 2012 with Remote Desktop Services (RDS) and Hyper-V. Microsoft VDI is comprised of session- based and VM-based solutions. For more information visit: <http://www.microsoft.com/en-us/windows/enterprise/products-and-technologies/virtualization/vdi.aspx>

Microsoft App-V

Microsoft Application Virtualization (App-V) enables enterprises to meet the needs of users and IT by empowering anywhere productivity and accelerated application deployment. App-V provides on-demand user access to applications on any authorized device without application installation. App-V increases business agility through faster application deployment and updates with no user interruptions. It minimizes conflicts between applications, allowing enterprises to reduce application compatibility testing time. For more information visit: <http://www.microsoft.com/en-us/windows/enterprise/products-and-technologies/virtualization/app-v.aspx>

Configuration Manager

Microsoft System Center 2012 Configuration Manager helps companies empower people to use the devices and applications they need to be productive, while maintaining corporate compliance and control. It accomplishes this with a unified infrastructure that gives a single pane of glass to manage physical, virtual, and mobile clients. It also provides tools and improvements that make it easier for IT administrators to do their jobs. With SP1, it provides integration with Windows Intune to manage PCs and mobile devices, both from the cloud and on-premise, from a single administrative console. For more information visit: <http://www.microsoft.com/en-us/server-cloud/system-center/configuration-manager-2012.aspx>

User State Virtualization

User State technologies transfer user data and settings to a centralized data center, eliminating the constraints of local storage and allow users to access their data and settings from any desktop (physical, session, or VM). User State technologies give users the flexibility to work everywhere, improving business continuity. Users who switch between physical desktops in the office to personal devices at home or to mobile devices on the road, need the ability to

customize their application environment on each device to avoid lost productivity. Today's computing environment needs to provide a seamless end-user experience that can move with users across their computing environments. RDS User Profile Disks provide state retention across hosts in a collection and Microsoft User Experience Virtualization (UE-V) provides state synchronization across devices, sessions, and VM's. Together, User Profile Disks and UE-V, provide pooled VDI (VM-based) with 'state' and synchronization.

This document does not provide guidance for user-state virtualization and desktop virtualization solutions, but does provides links to additional reference documentation that give further guidance for integrating into the complete solution.

User Profile Disks

Windows Server 2012 RDS introduces User Profile Disks as a specific user data management solution for collections (session- or pooled VM-based) in Microsoft VDI solutions. User Profile Disks create a per-collection and per-user virtual hard disk that stores changes to the user data and settings that can be recomposed on a different VDI (session-based) server in the farm or pooled VDI (VM-based). This reduces the need for Roaming Profiles that can utilize precious disk space on RDSH servers or on the SAN with pooled and personal VDI (VM-based). For more information visit: <http://technet.microsoft.com/en-us/library/hh831527.aspx>

Microsoft User Experience Virtualization (UE-V):

Microsoft User Experience Virtualization is an enterprise-scale user-state virtualization solution that delivers a user's personal Windows experience across many devices, is simple for IT to deploy, and easily integrates with existing management tools. UE-V enables application and Windows settings to roam across Microsoft VDI and physical desktops. UE-V enables users to change their device and keep their desktop experience (application and Windows settings) each time they login to a different desktop.

UE-V integrates with Windows Folder Redirection and Offline Files, providing data management and enabling roaming of documents, pictures, desktop, and application data, while maintaining the data in the secure and backed-up datacenter. Combining UE-V and User Profile Disks enables recomposing user personalization of desktops, whether physical or virtual. For more information visit: <http://www.microsoft.com/en-us/windows/enterprise/products-and-technologies/virtualization/UE-V.aspx>

Benefits of App-V and Microsoft VDI

Combining App-V and Microsoft VDI (session and VM-based) provides benefits that can accelerate the deployment and reduce the management and support efforts required. Although general App-V benefits apply with VDI, there are specific benefits that apply to VDI (session, VM, or both solutions).

VDI (Session based) Benefits

The VDI (session based) environment (many users sharing a single server or servers, but with individual desktops and applications), is different from a desktop environment (one user utilizing a single client OS resources and applications). The benefits of App-V translate into the VDI (session based) environment, and include:

- **Consolidate Servers and End Server Siloing, Increasing Server Farm ROI:** App-V's application virtualization allows most applications to run alongside any other—even applications that normally conflict, as well as multiple versions of the same application and many applications that previously could not run under Remote Desktop Services. This eliminates the need for server silos and significantly improves server utilization. As a result, organizations can consolidate their applications into a single farm, or just a few farms, for specific user needs, reducing the management and support efforts.
- **End Application Conflicts and Regression Testing:** Microsoft App-V for RDS prevents problems that hinder deployments by eliminating the need to permanently install applications on servers, and shielding the operating system and applications from changes created when running installed applications. The need to perform lengthy regression testing is also significantly reduced.
- **Accelerate Application Deployment:** Applications that use App-V typically only need to be packaged once and delivered to both desktop and Remote Desktop Services platforms. However, the packages should be tested on all target platforms to ensure compatibility. This reduces the need for "double packaging," or creating two different processes and packages when providing the choice of running an application on a desktop or via a Remote Desktop Session Host server.
- **Reduce Deployment Risk:** With Microsoft App-V, applications can be deployed on demand to users without disconnecting users and rebooting servers. Without App-V, installing a new application on a Remote Desktop Session Host server was not automated; installation required the draining of users so all were logged off, the RDSH server had to be changed to installation mode, and servers had to be rebooted to finish application installations.

VDI (VM based) Benefits

In addition to VDI (session based), RDS also provides a platform for a Virtual Desktop infrastructure (VM based). With RDS, administrators have a unified experience for setting up user access to VDI (session based) applications and desktops as well as virtual machine-based, isolated desktops in the data center. Leveraging App-V in a VDI environment provides similar benefits to the traditional desktop environment but also addresses some unique issues. These benefits include:

- **Simplify base images with dynamic application provisioning to clients:** Users connecting to virtual machines can access provisioned applications on demand. This allows for smaller and more flexible master images, as the application layer is assembled on top of the virtual machine. In addition, App-V reduces the network overhead by only delivering the requested portions of the applications. These portions remained cached by Windows for subsequent use during a user session with Shared Content Store mode.
- **Minimize application duplication:** One cost of dynamic provisioning is that applications need to be streamed to VDI (VM). This increases the launch time and requires additional disk space as each virtual machine that uses the application will have its own copy. For example, having 1,000 virtual desktops that each have a copy of Microsoft Office has a storage impact (1,000 VMs x 10GB = 10 TB) for Windows 8 and Office. To avoid this, the App-V Shared Content Store mode eliminates the local storage by streaming each application on demand when the user needs it. In the example above with 1,000 VDI (VM), the reduction of storage is approximately 2 TB (1,000 VMs x 8 GB= 8TB) without Microsoft Office stored in the VDI (VM). With shared content store, the applications are streamed from shared storage into each virtual machine provisioned (user or computer targets).. The on-disk footprint is reduced and the application is ready to run in advance of the user's request. Implementing Shared Content Store Mode does not increase the administrative process, and utilizes the same application provisioning management workflow.

Microsoft VDI with App-V Architecture

Building a Microsoft VDI with App-V solution requires an App-V Delivery infrastructure, Session Collection(s), Virtual Desktop Collection(s), and Client Access components. The primary RDS infrastructure that manages the collections and provides client access is the same whether the App-V applications are deployed from an App-V or Configuration Manager Infrastructure. The following diagram and table display and describe each component of an integrated solution.

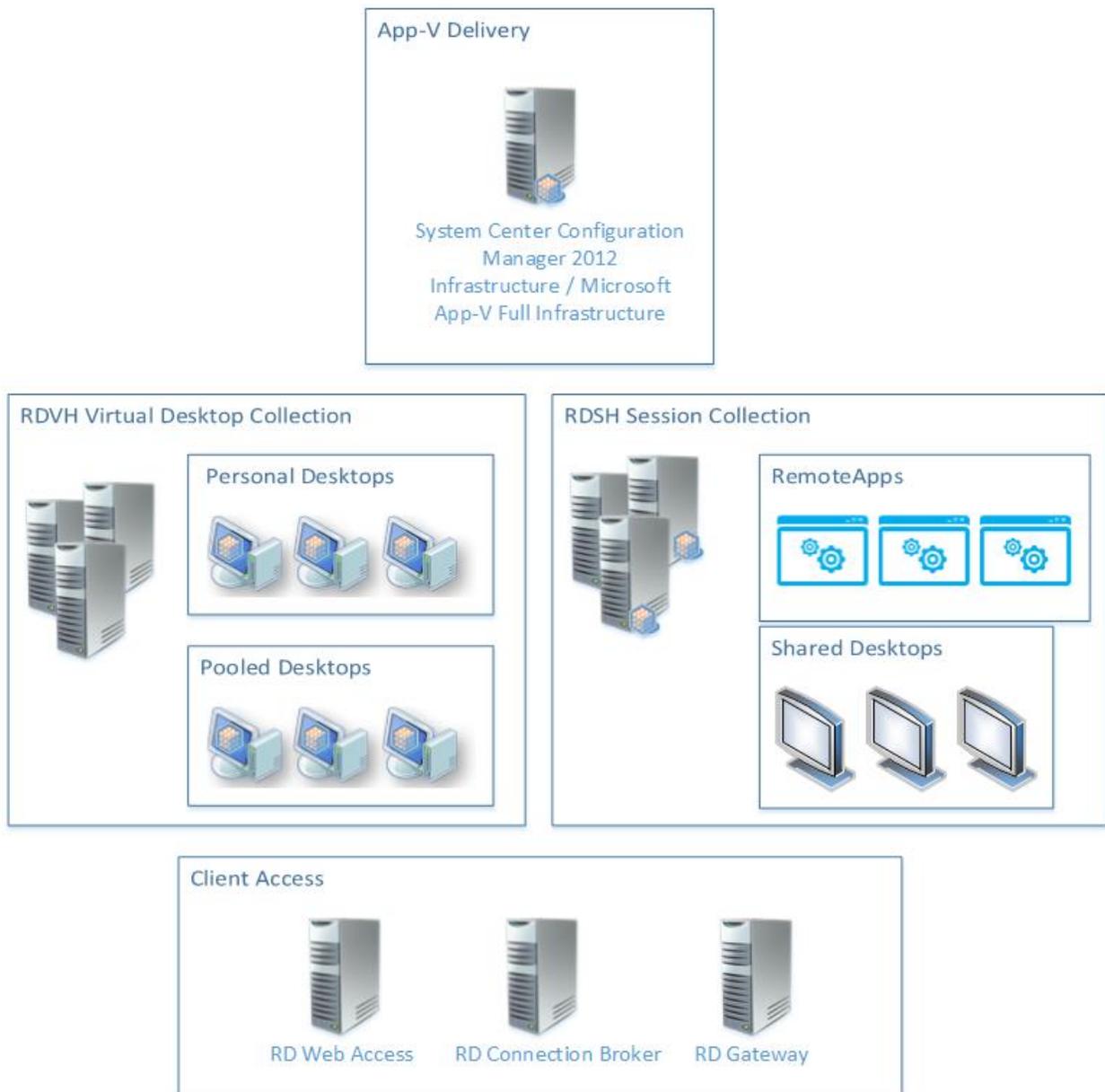


Figure 1: Microsoft VDI with App-V Infrastructure

The diagram above shows the components broken into App-V delivery infrastructure, virtual desktop and session collections, and client access infrastructure. The tables below explain specific infrastructures with descriptions of the components.

App-V Delivery Infrastructure

There are two specific App-V delivery infrastructures that support a Microsoft VDI solution. The following are the recommended set of components. Many of the components can be installed on either a single server, or a small number of servers. Please refer to the specific deployment guidance for both App-V Delivery infrastructures when building the supporting infrastructure.

App-V Infrastructure

App-V Management Server	The centralized management location for App-V infrastructures, providing authentication, security, metering, and monitoring from a Silverlight-based management site.
App-V Publishing Server	Provides App-V Clients with entitled applications for specific user and computers and hosts the package for streaming.
Microsoft App-V Sequencer	The App-V Sequencer packages applications into the App-V format for deployment with Configuration Manager or App-V full infrastructure.
App-V Client (Windows Desktop and Remote Desktop Services)	The App-V client manages the virtual applications launched by the user on the client device. The App-V client performs the publishing of virtual applications and the streaming of application data in the package store.

Configuration Manager Infrastructure

Configuration Manager Primary Site	A part of the Configuration Manager site hierarchy required for managing the distribution and deployment of applications to client machines.
Configuration Manager Distribution Point	A site system role responsible for hosting application content that is made available to Configuration Manager clients. Such data can include files for applications, packages, software updates, operating system images, boot images and drivers.
Configuration Manager Management Point	A site system role that manages a client's actions. It is responsible for relaying information to a client from the primary site. These actions include local machine policies and content locations.

Configuration Manager Application Catalog	A web portal used by end-users to request software that has been published by a Configuration Manager administrator. The catalog interacts with a Configuration Manager client to install software as soon as a client has requested it.
Microsoft App-V Sequencer	The App-V Sequencer packages applications into the App-V format for deployment with Configuration Manager.
Configuration Manger Client	The Configuration Manger client interacts with ConfigMgr server roles to facilitate the download of content and perform the installation of applications. For App-V, it instructs the App-V client to publish virtual applications and assists in locating App-V streaming content locations. Supported client devices include desktop/laptop PC's, session (Remote Desktop Session Host and Citrix XenApp™), and Virtual Desktop Infrastructure (VDI) clients.
App-V Client (Windows Desktop and Remote Desktop Services)	The App-V client manages the virtual applications launched by the user on the client device. The App-V client performs the publishing of virtual applications and the streaming of application data in the package store.

VDI (Session and VM-based) Collections

The RDS roles, Remote Desktop Session Host and Remote Desktop Virtualization Host, manage the provisioning of shared desktops, RemoteApp, and virtual desktops, where App-V applications are delivered. RDSH and RDVH provide user-level provisioning to control access to the App-V applications delivered from App-V or Configuration Manager 2012 SP1.

RDSH Servers	Hosts and provisions applications to shared desktop and RemoteApp programs managed logically in VDI (session based) collections.
RDVH Servers	Utilizes Hyper-V to host VDI (VM based) and logically manage collections of pooled or personal virtual desktops.

Client Access Infrastructure

The Microsoft VDI with App-V solution requires these RDS components: Remote Desktop Connection Broker and either RDSH or RDVH. In addition, internet connectivity and web publishing require Remote Desktop Gateway and Remote Desktop Web Access.

Remote Desktop Web Access	Provides access from web-connected clients via Internet Explorer and the RemoteApp and Desktop Connection feed.
Remote Desktop Gateway	Enables secure access from the internet to internal RDS hosted shared desktops, RemoteApp, and virtual desktops.
Remote Desktop Connection Broker	Manages farms of RDSH and RDVH servers, enabling reconnection of lost session, balancing user load, and facilitating virtual desktop startup, connections, and reset.

Common Components and Configurations

Across each of the deployment targets for App-V Applications: RDSH, pooled, and personal VDI (VM-based) there are common components and configurations that make up App-V, Configuration Manager, or RDS integrated solutions. This section will describe the details of each of these common components. The detailed configurations in the sections below will expand on this information and detail specific configuration differences for these common components. The following is a list of the common components that make up Microsoft VDI with App-V solutions:

- App-V Shared Content Store Mode
- RemoteApp and Connections

Shared Content Store

App-V 5.0 is designed to be easy and efficient to use in Microsoft VDI environments, allowing IT to make the best use of expensive disk resources without changing the user experience. App-V 5.0 Shared Content Store turns off local virtual application storage, dramatically reducing disk requirements for VDI while leaving the application provisioning and update process unchanged.

App-V Shared Content Store mode changes the default behavior of the App-V client so virtual applications are not cached on the local App-V Client. Rather, virtual applications stream into RAM on-demand at each launch of the application. Shared Content Store Mode provides flexibility where administrators can choose which applications to cache locally for highly utilized applications. Shared Content Store eliminates duplicating virtual applications across tens to hundreds of RDSH servers in a farm or hundreds to thousands of VDI (VM-based).

Shared Content Store mode can be configured during setup or after installation by modifying a registry value or from the built-in App-V Client PowerShell module. The following options are available for configuring Shared Content Store mode after installation.

Registry

From the registry editor navigate to **HKLM\Software\Microsoft\AppV\Client\Streaming** and edit **SharedContentStoreMode** to a value of 1.

PowerShell

Open PowerShell as an administrator in the RDSH or virtual desktop.

```
Import-Module AppvClient  
Set-AppvClientConfiguration -SharedContentStoreMode 1
```

Note: Configuring Shared Content Store Mode after installation requires a restart of the App-V Client Service (Windows Desktop or RDSH). Review the installation instructions in the [Appendix](#) section or view the TechNet documentation at: <http://technet.microsoft.com/en-us/library/jj713455.aspx> for configuring shared content store mode during installation.

For VDI (session-based) the application is streamed into memory on the first use and the Windows memory manager manages the in-memory cache of the application. Subsequent users will not have to stream the application again as long as the memory manager has not removed the virtual application from memory. Shared Content Store mode is a configuration of the App-V Desktop and RDS Client that can reduce disk space requirements on RDSH, pooled and personal desktops.

RemoteApp and Desktop Connections

Microsoft RemoteApp and Desktop Connections enables organizations to provide access to standard Windows-based programs from virtually any location to users with computers running Microsoft Windows 8, Windows 7, Windows Server 2012, or Windows Server 2008.

RemoteApp programs are accessed remotely through Remote Desktop Services and appear as if they are running on the end user's local computer. Instead of presenting to the user in the desktop of the Remote Desktop Session Host server, the RemoteApp program integrates with the client's desktop, running in its own resizable window with its own entry in the taskbar. Users can run RemoteApp programs side-by-side with their local programs. If a user is running more than one RemoteApp program on the same Remote Desktop Session Host server, the RemoteApp programs will share the same Remote Desktop Services session.

RDS RemoteApp Desktop Connection Feed

RemoteApp and Desktop Connections works with the Remote Desktop Web Access (RD Web Access) feature, RemoteApp and Desktop Connection feed. Instead of presenting RemoteApp programs in the form of a web page, distributing RDP files, or installing MSI, this feed presents

applications in a software-parsable XML document. With RemoteApp and Desktop Connections, the user subscribes to a feed of RemoteApp programs by supplying the connection feed URL into the built-in operating system configuration or with the new Windows 8 Modern UI Remote Desktop app. The user subscribes to the feed, and then the RemoteApp and Desktop Connections client software places the resources (RemoteApps and Virtual Desktops) in the connection feed in the user's Start menu and updates periodically.

The RemoteApp and Desktop Connections feature offers several benefits:

- RemoteApp programs launch from the Start menu just like any other application.
- Published Remote Desktop connections are included alongside RemoteApp programs on the Start menu.
- Changes to the published connection (such as newly published RemoteApp programs) automatically reflect on the user's Start menu, without any effort on the part of the user.
- RemoteApp programs launch easily with Windows Search.
- Users only have to log on once, to create the connection. From that point on, updates happen with no prompt for user credentials.
- RemoteApp and Desktop Connections does not require client domain membership for client computers.
- RemoteApp and Desktop Connections benefits from features in RDS, such as Personal Desktop assignment or per-user application filtering.
- RemoteApp and Desktop Connections is built on standard technologies such as XML and HTTPS, making it possible for developers to build solutions around it. It also offers APIs that allow the client software to support other types of resources, in addition to RemoteApp programs and Remote Desktop connections.

This feature enables administrators designing the virtual desktop (pooled and personal) to configure the virtual desktop client to subscribe to an appropriate feed that has all of the necessary applications presented without footprint, and automatically updated from the centralized changes.

Integrating Microsoft VDI with App-V 5.0

Previous sections of this document outline the benefits provided by combining Microsoft VDI with App-V. The architecture includes three primary components covered in this whitepaper, App-V Deployment, Collection management, and Client Access. As mentioned previously, implementing a user-state technology to manage user data and settings complete the solution. The implementation of user-state technology is not part of the document as it is outside the scope of this document.

The next sections detail the implementation of App-V in Microsoft VDI solutions, including the specific differences between RDSH (shared desktop and RemoteApp) and virtual desktop (pooled and personal) solutions. App-V deployment with App-V or Configuration Manager infrastructures are presented to assist in planning and implementation. Although the documentation presents these as separate tasks, most organizations employ a combination of VDI (session- and VM-based).

Integrating Microsoft VDI with App-V can be accomplished with the sequencer-created MSI and with the PowerShell App-V module. Although they could be automated with scripting, this document will only focus on App-V and Configuration Manager infrastructure-based delivery. PowerShell uses will be mentioned in the document as steps for specific technical solutions where App-V and Configuration Manger infrastructure are not the correct choices.

Integrating App-V with RDSH

Remote Desktop Session Host, formerly known as Terminal Services, is a proven and mature centralized desktop infrastructure that many organizations have already deployed, and it has the greatest user density and minimum hardware requirements in the Microsoft VDI stack. Windows Server 2012 makes it easier to deploy this architecture by offering a VDI (session based) deployment scenario-based installation.

Organizations leverage RDS Sessions to provide access to applications from a collection of RDSH servers (from one to many) to users with a compatible Remote Desktop Connection client. IT departments implement App-V to save costs by leveraging the reduction of testing and deployment time, and utilize the isolation feature of App-V to eliminate application conflicts, enabling consolidation of RDSH server silos. This enables a common collections of servers (farm) that host the targeted applications for an entire organization. Applications can be delivered to RDSH hosts with ConfigMgr or App-V Full Infrastructure and made available to users via a shared desktop or RemoteApp session.

Applications on RDSH solutions enable organizations to deliver applications in several ways. As previously mentioned, users can consume applications as a shared desktop with published applications, or as entitled RemoteApps. Because the Windows 7 and Windows 8 operating systems have built-in RemoteApp and Desktop Connections features, they offer additional delivery scenarios across the RDS service offerings that appear in the start menu like traditional applications.

This section will focus on enabling delivery of App-V applications and entitlement in a RDSH-based solution. Infrastructure components in the diagram and tables in the [Microsoft VDI with App-V Architecture](#) section should be reviewed to build the solution.

Beyond the RDS and Deployment infrastructures, the following client-based components are required:

App-V Infrastructure	<ul style="list-style-type: none">• RDSH Role: Must be installed prior to App-V Client• App-V RDS Client
Configuration Manager Infrastructure	<ul style="list-style-type: none">• RDSH Role: Must be installed prior to App-V Client• Configuration Manager Client• App-V RDS Client

Deploying the RDSH Role

Including App-V on an RDSH server requires specific considerations for a complete solution. To begin, install the RDSH role utilizing the instructions provided in the link:

<http://technet.microsoft.com/en-us/library/hh831447>

Deploying the App-V RDS Client

When deploying the App-V RDS Client, ensure that the RDSH role has been installed. After the RDSH role has been installed, the **App-V Client for RDS (appv_client_setup_rds.exe)** must be installed on the RDSH server. Some important factors when installing the App-V Client for Remote Desktop Service are:

Configuration of the App-V Client

There are several configurations to support RDSH with App-V deployments during or post-installation:

- **Location of the App-V package store:** Place the package store (default c:\programdata\appv) on a drive with adequate storage for all virtual applications targeted for delivery.
- **Publishing Server (App-V Management):** Implementations with an App-V Management Infrastructure require a publishing server configuration. This is not required for Configuration Manager-based management.
- **Pre-Caching App-V Applications (Recommended):** Ensure that App-V application assets are placed in their entirety in the local App-V package store. The App-V Client provides configuration options to automatically cache applications (autoload) that have been published to clients. In the absence of this configuration, PowerShell can be utilized to pre-cache applications.
- **Global Refresh (Recommended):** Enables computer-based targeting of virtual applications to RDSH servers where an App-V Management Infrastructure is utilized. Global Refresh provides automated integration of applications for RemoteApp publishing.
- **Enable Package Scripts (Recommended):** Some packages include scripting to achieve optimal operation. Enable package scripts during setup or post-installation, unless it is certain that no packages will use scripts.
- **Shared Content Store Mode (Optional):** Reduces the storage requirements across the entire RDSH Farm.

For instructions on how to deploy the App-V Client via Configuration Manager, see the [Deploying the App-V Client via Configuration Manager](#) section of this document.

For instructions on how to deploy the App-V Client manually, see the [Deploying the App-V Client Manually](#) section of this document.

Virtual Application Deployment to Remote Desktop Servers

When planning a RDSH server deployments with App-V, there are two primary manners in which users consume the applications; shared desktops and RemoteApp/web access. RDSH is the delivery and execution point for the applications, but the way users consume determines how App-V application deliveries are performed.

The following table lists the supported publishing options and recommended Remote Desktop Services features (Remote Desktop and RemoteApp) when used with different App-V deployment methods:

RDSH Consumption	Deployment Method	Shared Content Store Mode	Settings and Configurations	Target (user/computer)
Shared Desktop*	App-V	No	<ul style="list-style-type: none"> Autoload 2 or manual pre-cache SharedContentStoreMode 0 	User and Computer
		Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 	
	Configuration Manager	No	<ul style="list-style-type: none"> Autoload 2 SharedContentStoreMode 0 Streaming or Download and Run Locally deployments 	
		Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 Streaming deployment 	
RemoteApp/ WebAccess**	App-V	No	<ul style="list-style-type: none"> Autoload 2 or manually pre-cache SharedContentStoreMode 0 Global publishing Requires Publishing of App-V application as RemoteApp/WebAccess 	Computer
		Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 Requires Publishing of App-V application as RemoteApp/WebAccess 	

	Configuration Manager	No	<ul style="list-style-type: none"> Autoload 2 or manually pre-cache SharedContentStoreMode 0 Machine targeting Requires Publishing of App-V application as RemoteApp/WebAccess Streaming deployment
		Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 Machine targeting Requires Publishing of App-V application as RemoteApp/WebAccess Streaming deployment

Table 1: Virtual Application Deployment to Remote Desktop Session Host Servers

The table above describes possible methods for deploying virtual application packages to RD Session Host servers.

*When utilizing Shared Desktops, the use of user-based targeting is preferred, as users will only be presented with the App-V applications that have been published to them.

**Publishing of App-V applications as RemoteApp/WebAccess could also be automated with PowerShell and implemented with App-V Dynamic Configuration files, or with post deployment tasks with Configuration Manager. The writing of these scripts is outside the scope of this document.

Reduce RDSH Storage Requirements

Organizations with RDSH farms can reduce the storage requirements associated with having the same applications copied to multiple servers using the App-V Shared Content Store Mode configuration feature. This configuration changes the App-V Client cache behavior, streaming applications to RAM on launch instead of the package store. The virtual application's bits are streamed as needed, from either the Configuration Manager Distribution Point or the App-V Publishing Server over the network. The bits are streamed into RAM at the first user launch after reboot, where the in-memory cache of the virtual application is managed by the Windows memory manager.

Note: Organizations can create start-up scripts on their RDSH servers that will launch and then close the most frequently used applications. This will place the application run-times into memory on the server and will provide a better user experience as the frequently launched applications will already be in memory. Start-up scripts are limited by how much available RAM is on each RDSH server and how the Windows Memory Manager handles each subsequent

launch, so testing is recommended. The creation of startup scripts is outside the scope of this document.

Streaming deployments are recommended for Configuration Manager App-V application deployments leveraging the App-V Shared Content Store. This will ensure that the virtual application is not cached on the local machine. For environments where specific applications need to be placed into the package store (cache) to improve response times, Download and Run Locally will place the App-V package in the Configuration Manager Cache and automatically stream the virtual application into the package store. For Configuration Manager deployment solutions to Shared Content Store mode clients and streaming deployments, it is recommended that the location of the distribution points are limited to local network for optimized launch performance.

Deploying App-V Applications to RDSH using Configuration Manager

1. The application is sequenced and an App-V Virtual Application Package (.appv file) is created.
2. An App-V Deployment Type is created for the App-V Application and is copied to the appropriate distribution points.
3. A Deployment is created targeting either user- or machine-based collections by way of either "Streaming" or "Download and Run".

Reminder: The "Download and Run" deployment method instructs the client to locally cache the application prior to first launch. Environments that are utilizing App-V Shared Content Store mode should utilize the "Streaming" method, unless there are specific applications that need to be locally cached (improving launch time for larger packages like Office).

Note: For more details about modeling App-V virtual application packages in Configuration Manager download and review: <http://go.microsoft.com/?linkid=9824871>

Deploying App-V Applications to RDSH using App-V Full Infrastructure

1. The application is sequenced and an App-V Virtual Application Package (.appv file) is created.
2. The Virtual Application Package is imported on the App-V Management server.
3. The Virtual Application Package is published to the appropriate AD groups on the App-V Management Server.

Note: For detailed information on deploying applications with App-V Full Infrastructure review the steps in the App-V 5 Trial Guide at: <http://www.microsoft.com/en-us/download/details.aspx?id=27760>

Publishing App-V apps with RemoteApp

Publishing for RemoteApp/WebAccess is a separate process from publishing the App-V application on the RDSH server. This process could be automated with the use of App-V Dynamic Configuration files or PowerShell scripting. This process enables administrators to choose which users are to be targeted for consumption of the application.

1. Open **Server Manager** and select **Remote Desktop Services** and then **Collections**.
2. Browse to the relevant **RDS Session collection**.
3. In the **REMOTEAPP PROGRAMS** pane, click **Tasks** and select **Publish RemoteApp Programs**.
4. The globally published App-V application will be displayed in the list of available applications. Select the App-V application(s) from the list and complete the publishing wizard.

Note: It is recommended that App-V applications are published only to computer targets (RDSH Servers) if they are consumed as RemoteApps.

Subscribing to a RemoteApp and Desktop Connections Feed

After all of the RemoteApps have been published, a user can subscribe to the RemoteApp and Desktop Connections feed to have the applications presented in the operating system from the Start Menu. The RemoteApp and Desktop Connections feed is periodically updated, which enables users of a desktop with an active feed connection to receive an updated list of RemoteApps or virtual desktops. RemoteApp and Desktop Connections feeds can be subscribed to from both domain (managed) and non-domain (unmanaged) desktops.

Managed (Domain joined)

Administrators can configure Group Policy Objects (GPO) to assign the address of the feed. This configuration, combined with the single sign-on capabilities of RDS, allows users to be automatically enrolled and authenticated. The end result is a start menu (Windows 7) or start screen (Windows 8) that is populated with RemoteApps and virtual desktops that have been provisioned for the specific user.

Perform the following tasks:

1. Utilize **Group Policy Management Console** or **Advanced Group Policy Management** to create a new GPO.
2. Target or link the GPO to an organizational unit that has the appropriate users. Apply security or WMI-based filters, if required, to refine the scope.

3. From the Group Policy Editor go to **User Configuration | Policies | Administrative Templates | Windows Components | Remote Desktop Services | RemoteApp and Desktop Connections**
4. Select the setting on the right called **Specify default connection URL** and edit.
5. **Enable** the setting and type the location of the **RemoteApp and Desktop Connection** feed **http(s)://{FQDN of RD Web Server}/RDweb/feed/webfeed.aspx**.

Note: This configuration may have to be altered if utilizing different domain names for internal and external DNS. Also, if utilizing a secure connection feed location, the name specified in the DNS record must match the certificate name since the Modern UI Remote Desktop client doesn't prompt for name mismatch errors. For additional information on configuring digital signatures for RemoteApp programs visit:

<http://technet.microsoft.com/en-us/library/cc754499.aspx>

Unmanaged (non-domain joined)

Windows 8 supports finding the connection URL when utilizing the Modern UI Remote Desktop client. In this scenario, a user opens the Modern UI Remote Desktop app and chooses to **Access RemoteApp and Desktop Connections**. From the entry field they can specify the connection URL if known, or simply specify their email address. This configuration enables users to access RemoteApp and Desktop Connections from unmanaged computers and requires the following configuration.

1. Open DNS Manager and connect to the DNS server that manages the correct namespace.
2. Right-click the forward lookup zone of the domain, and select **Other New Records**.
3. Click **Text (TXT)**, then click **Create Record**.
4. In the Record name box, type "**_msradc**".
5. In the Text box, type **http(s)://{FQDN of RD Web Server}/RDWeb/Feed/webfeed.aspx**.
6. Click **OK**
7. Close DNS Manager.

Note: This configuration may have to be altered if utilizing a different domain name for the email address than the internal DNS domain name. Also, if utilizing a secure connection feed location, the name specified in the DNS record must match the certificate name since the Modern UI Remote Desktop client doesn't prompt for name mismatch errors.

Integrating App-V in Personal VDI (VM based) Collections

Organizations can leverage Personal virtual machines to provide access to desktops from any location to users requiring a persistent desktop. Since virtual machines are often used as a desktop replacement and the data, applications and settings are assigned to them indefinitely, the ability to reduce the size of the storage footprint of the desktop is critical. Integrating App-V for its typical desktop benefits is the first step, but enabling Shared Content Store mode is nearly as important to reduce storage requirements on the expensive SAN.

In this scenario, the App-V Client experience is similar in virtual desktops as it is with Physical Desktops. App-V applications are delivered and managed from Configuration Manager or App-V Management Infrastructure, with similar tasks. Depending on the management infrastructure, the following components make up the personal virtual desktop solution.

Beyond the RDS and App-V Deployment infrastructures, the following client based components are required:

App-V Infrastructure	<ul style="list-style-type: none">• App-V Desktop Client
Configuration Manager Infrastructure	<ul style="list-style-type: none">• Configuration Manager Client• App-V Desktop Client

Deploying the App-V Client

Depending on the management infrastructure, the App-V Client can be deployed to the master image or, after the personal desktop has been provisioned, to a specific user. There is not a process for deploying the client with App-V management infrastructures, so it is recommended that the client is installed and captured in the master image. For Configuration Manager solutions, it is recommended that the App-V Client and ConfigMgr Client be installed as part of the master image to ensure the assigned desktop is ready to use immediately after provisioning.

Configuration of the App-V Client

During or post-installation, there are several configurations to support pooled VDI (VM based) with App-V deployments:

- **Location of the App-V package store:** Place the package store (default c:\programdata\appv) on a drive with adequate storage for all virtual applications targeted for delivery.
- **Publishing Server (App-V Management):** Implementations with an App-V Management Infrastructure require a publishing server configuration. This is not required for Configuration Manager-based management.

- **Disable Pre-Caching App-V Applications (Recommended):** Ensure that App-V application assets are not placed in their entirety in the local App-V package store. The App-V Client provides configuration options to disable automatically-cached applications (autoload). This is not required when Shared Content Store mode is enabled. Shared Content Store mode disables the use of the Autoload settings.
- **Global Refresh:** Enable computer-based targeting of virtual applications to personal virtual desktops where an App-V Management Infrastructure is utilized.
- **Enable Package Scripts (Recommended):** Some packages include scripting to achieve optimal operation. During setup or post-installation, enable package scripts, unless it is certain that no packages will use scripts.
- **Shared Content Store Mode (Recommended):** Reduce the storage requirements across the entire personal virtual desktop collections.

For instructions on how to deploy the App-V Client directly, see the [Deploying the App-V client Manually](#) section of this document.

For instructions on how to deploy the App-V Client via Configuration Manager, see the [Deploying the App-V Client via Configuration Manager](#) section of this document.

For instructions on how to deploy the Configuration Manager Agent to desktop clients, see Microsoft TechNet at: <http://technet.microsoft.com/en-us/library/gg699391.aspx>

Virtual application Deployment to Personal VDI (VM based)

Publishing App-V applications in personal VDI (VM-based) can be accomplished in the same way as delivery to physical desktops. However, since personal VDI (VM-based) are stored on expensive SAN storage, it is recommended to enable Shared Content Store Mode. This is not absolute however, and organizations may choose to evaluate the storage impact based on the size of the virtual applications that are planned for deployment, multiplied by the number of desktops to make a final decision.

Collection Type	Deployment Method	Shared Content Store Mode	Specific Settings and Configurations	Target (user/computer)
Personal Desktops	PowerShell (pre-stage)	No	<ul style="list-style-type: none"> Place typical LOB applications into the master image 	Computer
		Yes		Computer
	App-V	No	<ul style="list-style-type: none"> Autoload 1 or 2 SharedContentStoreMode 0 	User and Computer
		Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 	User and Computer
	Configuration Manager	No	<ul style="list-style-type: none"> Autoload 1 or 2 SharedContentStoreMode 0 Streaming or Download and Run Locally deployments 	User and Computer
		Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 Streaming deployment 	User and Computer
	RemoteApp and Desktop Connections	NA	<ul style="list-style-type: none"> GPO Delivery of connection feed 	User GPO Settings

Reduce Personal VDI (VM based) Storage Requirements

Organizations with personal VDI (VM-based) can reduce the storage requirements associated with having the same applications copied to multiple virtual desktops using the App-V Shared Content Store configuration feature. This configuration changes the App-V Client cache behavior, streaming applications on launch to RAM instead of the package store. The virtual application's bits are streamed as-needed, from either the Configuration Manager Distribution Point or the App-V Publishing Server over the network, and into RAM at the first user launch after reboot, where the in-memory cache of the virtual application is managed by the Windows memory manager.

App-V Management Infrastructure deployments should ensure that the autoload setting for the App-V client is set to 0 (none) which disables caching applications to the package store.

Streaming deployments are recommended for Configuration Manager App-V application deployments leveraging the App-V Shared Content Store, which will ensure that the virtual application is not cached on the local machine. For environments where specific applications need to be placed into the package store (cache) to minimize launch times, Download and Run Locally will place the App-V package in the Configuration Manager Cache and automatically stream the virtual application into the package store. For Configuration Manager deployment solutions to Shared Content Store mode clients and streaming deployments, it is recommended that the location of the distribution points are limited to local network for optimized launch performance.

Deploying App-V Applications to personal VDI (VM based) using Configuration Manager

1. The application is sequenced and an App-V Virtual Application Package (.appv file) is created.
2. An App-V Deployment Type is created for the App-V Application and is copied to the appropriate distribution points.
3. A Deployment is created targeting either user or machine based collections by way of either "Streaming" or "Download and Run".

Reminder: The "Download and Run" deployment method instructs the client to locally cache the application prior to first launch. Environments that are utilizing App-V Shared Content Store mode should utilize the "Streaming" method, unless there are specific applications that need to be locally cached (improve launch time for larger packages like Office).

Note: For more details about modeling App-V virtual application packages in Configuration Manager download and review: <http://go.microsoft.com/?linkid=9824871>

Deploying App-V Applications to personal VDI (VM based) using App-V Full Infrastructure

1. The application is sequenced and an App-V Virtual Application Package (.appv file) is created.
2. The Virtual Application Package is imported on the App-V Management server.
3. The Virtual Application Package is published to the appropriate AD groups on the App-V Management Server.

Note: For detailed information on deploying applications with App-V Full Infrastructure review the steps in the App-V 5 Trial Guide at: <http://www.microsoft.com/en-us/download/details.aspx?id=27760>

Alternate Application Delivery to VDI (VM based)

Combining VDI (session and VM-based) simplifies the delivery and eliminates the storage requirements associated with delivering applications to either personal or pooled VDI (VM based). Organizations combine VDI (Session based) solutions with RemoteApp to provide delivery of the applications, App-V or traditional, to pooled or personal VDI (VM based) with a RemoteApp and Desktop Connections feed. This combination eliminates the storage requirements in the VDI (VM based) desktop for applications, and since the applications are already delivered to the RDSH servers and the RDSH Servers are typically on the same high-speed backbone network, the performance of RemoteApp applications will meet user expectations. Please review the information in this document in the [RemoteApp and Desktop Connections](#) section.

Integrating App-V in Pooled Virtual Machine Collections

When users do not require the ability to persist information between sessions (task-based workers,) organizations leverage pooled virtual machines to provide access to desktops from any location with increased density over personal desktop. Organizations often use the virtual machine as a desktop replacement; therefore, all required applications multiplied by the number of desktops creates a large storage requirement. Integrating App-V for its typical desktop benefits is the first step, but enabling Shared Content Store mode is nearly as important to reduce storage requirements on the expensive SAN.

In this model, the desktop resets after each user disconnects, requiring some specific deployment scenarios that make pooled VDI (VM-based) unique. In this model, all users in the virtual machine pool share a single master image. The changes that each user makes during a session are stored in a transient virtual hard disk that is discarded when the user logs off (though user profile changes can be persisted via User Profile Disks and/or UE-V as described earlier).

This combination of pooled VDI (VM-based) and user-state virtualization allows organizations to provide a more personal experience without the costs of maintaining and running one virtual machine for each user. Only a pool of enough virtual machines to support user session demand is required, while delivering a similar experience to personal VDI (VM based). Since different business units require different applications, multiple master images and pooled VDI (VM-based) collections must be created and maintained without App-V.

App-V provides pooled VDI (VM based) with the ability for dynamic application delivery. This enables creation of fewer master images and collections since applications can be delivered on-

demand to the user after they have logged into the pooled virtual machine. This section outlines how App-V applications integrates into pooled VDI (VM based)

Beyond the RDS and App-V Deployment infrastructures, the following client-based components are required:

App-V Infrastructure	<ul style="list-style-type: none"> • App-V Desktop Client
Configuration Manager Infrastructure	<ul style="list-style-type: none"> • Configuration Manager Client • App-V Desktop Client

Deploying the App-V Client

The App-V Client operates from within the pooled VDI (VM-based) in the same way as a physical desktop, but with one significant difference. Since pooled VDI (VM-based) are stateless, installation of the App-V Client in the master image for the collection is required. Otherwise, the client must be installed each time a user connects to a pooled VDI (VM-based) session. In addition, the virtual applications will have to be published every time a user logs on to the pooled VDI (VM-based), unless the administrator has pre-staged the publishing of applications prior to creating the master image.

Configuration of the App-V Client

During or post-installation there are several configurations to support pooled VDI (VM-based) with App-V deployments:

- **Location of the App-V package store:** Place the package store (default c:\programdata\appv) on a drive with adequate storage for all virtual applications targeted for delivery.
- **Publishing Server (App-V Management):** Implementations with an App-V Management Infrastructure require a publishing server configuration. This is not required for Configuration Manager based management.
- **Disable Pre-Caching App-V Applications (Recommended):** Ensure that App-V application assets are not placed in their entirety to the local App-V package store. The App-V Client provides configuration options to disable automatically cache applications (autoload).
- **Enable Package Scripts (Recommended):** Some packages include scripting to achieve optimal operation, during setup or post-installation. Enable package scripts, unless it is known that no packages will use scripts.
- **Shared Content Store Mode (Required):** Reduce the storage requirements across the entire pooled VDI (VM-based) collection.

For instructions on how to deploy the App-V Client directly, see the [Deploying the App-V client Manually](#) section of this document.

For instructions on how to deploy the App-V Client via Configuration Manager, see the [Deploying the App-V Client via Configuration Manager](#) section of this document.

For instructions on how to deploy the Configuration Manager Agent to desktop clients, see Microsoft TechNet at: <http://technet.microsoft.com/en-us/library/gg699391.aspx>.

Virtual Application Publishing to Pooled VDI (VM-based)

Publishing App-V applications in pooled VDI (VM-based) can be accomplished in the same way as delivery to physical desktops. However, since pooled VDI (VM-based) discards user-based changes after disconnect and are stored on expensive SAN storage, enabling Shared Content Store Mode is recommended. This recommendation is not absolute and organizations can evaluate the storage impact by multiplying the size of the planned virtual applications by the number of desktops. The result may prove useful in making a final feasibility determination for pre-staging (applications in master image) or dynamically delivering (application delivery post connection) applications.

Pre-staging some applications during the creation of the master image still allows adding and updating applications when users log on. Typically, in pre-staging the master image updates are scheduled when any updates applied after the initial creation are applied to the default applications during the master image update. Whether pre-staging App-V applications or dynamically delivering all App-V applications, Shared Content Store Mode is recommended.

Collection Type	Deployment Method	Shared Content Store Mode	Specific Settings and Configurations	Target (User/Computer)
Pooled Desktops	PowerShell (pre-stage)	No	<ul style="list-style-type: none"> Place typical LOB applications into the master image 	Computer
		Yes		Computer
	App-V	Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 	User
	Configuration Manager	Yes	<ul style="list-style-type: none"> SharedContentStoreMode 1 Streaming deployment 	User
	RemoteApp and Desktop Connections	NA	<ul style="list-style-type: none"> GPO Delivery of connection feed 	User GPO Settings

Reduce Pooled VDI (VM-based) Storage Requirements

Organizations with pooled VDI (VM-based) can reduce the storage requirements associated with having the same applications copied to multiple virtual desktops by using the App-V Shared Content Store configuration feature. This configuration changes the App-V Client cache behavior, streaming applications to RAM instead of the package store on launch. The virtual application's bits are streamed over the network as needed, from either the Configuration Manager Distribution Point or the App-V Publishing Server, and into RAM at the first user launch after reboot, where the in-memory cache of the virtual application is managed by the Windows memory manager.

To ensure that the virtual application is not cached on the local machine for Configuration Manager App-V application deployments leveraging the App-V Shared Content Store, it is recommended to use streaming deployments. For environments where specific applications are to be placed into the package store (cache) to minimize launch times, pre-stage the App-V applications in the master image for the collection. For optimized launch performance of Configuration Manager deployment solutions to Shared Content Store mode clients and streaming deployments, it is recommended that the location of the distribution points are limited to local network.

Deploying App-V Applications to pooled VDI (VM-based) using Configuration Manager

1. The application is sequenced and an App-V Virtual Application Package (.appv file) is created.
2. An App-V Deployment Type is created for the App-V Application and is copied to the appropriate distribution points.
3. A Deployment is created, targeting user-based collections via "Streaming".

Note: For more details about modeling App-V virtual application packages in Configuration Manager download and review: <http://go.microsoft.com/?linkid=9824871>

Deploying App-V Applications to pooled VDI (VM-based) using App-V Full Infrastructure

1. The application is sequenced and an App-V Virtual Application Package (.appv file) is created.
2. The Virtual Application Package is imported on the App-V Management server.
3. On the App-V Management Server the Virtual Application Package is published to the appropriate AD user groups.

Note: For detailed information on deploying applications with App-V Full Infrastructure review the steps in the App-V 5 Trial Guide at: <http://www.microsoft.com/en-us/download/details.aspx?id=27760>

Alternate Application Delivery to VDI (VM-based)

Combining VDI (session- and VM-based) solutions simplifies the delivery and eliminates the storage requirements associated with delivering applications to either personal or pooled VDI (VM-based). Organizations combine VDI (Session-based) solutions with RemoteApp to provide delivery of applications, App-V or traditional, to pooled or personal VDI (VM-based), with a RemoteApp and Desktop Connections feed. This combination eliminates the storage requirements in the VDI (VM-based) desktop for applications. Since the applications are already delivered to the RDSH servers and the RDSH Servers are typically on the same high-speed backbone network, the performance of RemoteApp applications will meet user expectations. Please review the information in this document in the [RemoteApp and Desktop Connections](#) section.

Conclusion

As the industry continues moving towards the user-centric model and embracing more flexible work-styles, it is looking to VDI technologies to fulfill the challenge in ways that are flexible, scalable and cost effective. Microsoft Application Virtualization plays a major role in completing the user-centric solution, improving the reach of the benefits of App-V to VDI (session and VM-based) solutions.

Integrating App-V with Microsoft VDI allows for the consolidation of hardware resources and significant reductions in storage and application support costs, thanks to application isolation, repair, and the Shared Content features of App-V.

Integrating App-V with Microsoft VDI extends the flexibility of VDI to transform applications into true user-centric services that enable a user's complete application experience to follow them continuously, on-demand, between physical and virtual environments, while simplifying management.

Combining App-V with a hybrid of the Microsoft VDI solutions (both session- and VM-based) enables administrators to provide a consolidated and centrally-managed solution for organizations by utilizing the built-in App-V management functionality, or increased and consolidated management when utilizing Configuration Manager 2012 SP1.

Appendix

This section provides additional step-by-step documentation, referenced in each section above, to reduce redundancy.

Deploying the App-V Client via Configuration Manager

An installation model for the App-V 5.0 client is available publicly from Microsoft as part of the Configuration Manager 2012 SP1 SDK at: <http://go.microsoft.com/fwlink/?LinkId=276784>.

This model provides administrators a way to properly install the **App-V 5.0 Desktop Client** and all required prerequisites in the proper manner. This model includes three different applications:

- App-V 5.0 Client
- KB2533623
- WMF 3.0 (PowerShell 3.0)

KB2533623 and WMF 3.0 are prerequisites required only on Windows 7 and Windows Server 2008 R2. Windows 8 and Windows Server 2012 require no additional software to install the App-V client.

Steps for Using the Application Model Kit

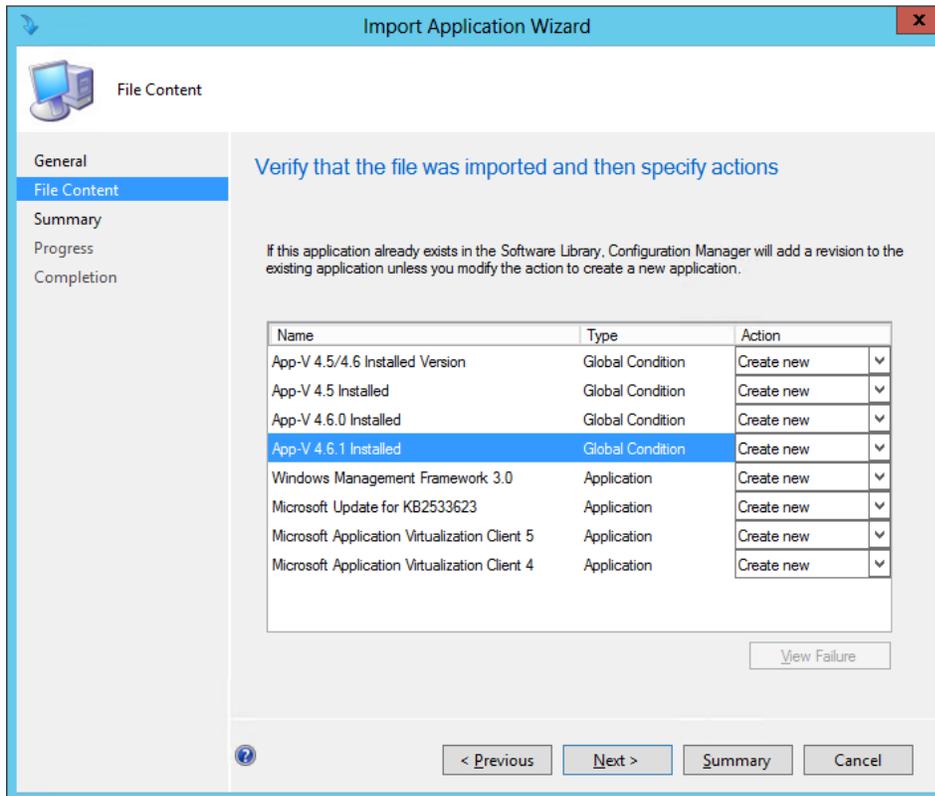
The application model provided by Microsoft contains only the required Configuration Manager objects. All source files required by the kit must be downloaded and configured after importing the clients. Delivering the App-V Desktop client to Microsoft VDI should only be performed for Personal VDI (VM-based). Pooled VDI (VM-based) images include the installed App-V Client. The following outlines the process:

1. Download the Configuration Manger SDK SP1 from <http://go.microsoft.com/fwlink/?LinkId=276784>.
2. Install the SDK.
3. Navigate to **C:\Program Files (x86)\Microsoft System Center 2012 Configuration Manager SDK\Samples\Application Model** (default location, may vary depending on installation options).
4. Copy the **Application Model Kit.zip** to machine with the ConfigMgr console installed.
5. Download the **App-V 5.0 Client** to a shared location.
6. Open the Configuration Manager console.
7. Navigate to **Software Library > Applications**.
8. Right click **Applications**, and select **Import Application**.
9. Click **Browse** and select the ZIP file copied in Step 4.

Note: The path must be in a UNC format.

1. Click **Next**.

Note: The wizard will analyze the contents of the ZIP file.



Notice the Applications and Global Conditions are contained within the ZIP.

2. Click **Next**.
3. Click **Next**. Click **Close**.
4. Select the Microsoft **Application Virtualization Client 5** application.
5. In the bottom frame, select the **Deployment Types** tab.
6. Right click the **Microsoft Application Virtualization (App-V) 5.0 Client Install from Script** deployment type, select Properties.
7. Select the **Content** tab.
8. Next to **Content Location** click **Browse**. Browse to the location setup in Step 2.
9. Click Select Folder.
10. Click **OK**.

The application source has now been updated. Repeat these steps for the WMF framework 3.0 and Microsoft Update KB2533623. Each deployment type contained within those applications will require the content location be updated. Once complete, the App-V 5.0 client is ready to be deployed.

Extending the Application Model Kit for deploying the App-V Client for RDS

To properly deploy the RDS App-V client to RDS server, a new deployment type needs to be created within the App-V 5 Client application. This deployment type will only run on a server with the Remote Desktop Session Host role installed. Additionally this deployment type will turn on Shared Content Store and Package Scripts by default, eliminating additional post-install steps.

The first step in the process is to create two new Global Conditions that will detect the presence of the Remote Desktop Session Host role on a server.

1. Navigate to the **Software Library** workspace, select **Global Conditions**.
2. From the ribbon, select **Create Global Condition**.
3. Fill out the following information:
 - a. Name: **RDSH Installed Query**
 - b. Device Type: **Windows**
 - c. Condition Type: **Setting**
 - d. Setting Type: **WQL Query**
 - e. Data Type: **String**
 - f. Namespace: **root\cimv2**
 - g. Class: **Win32_ServerFeature**
 - h. Property: **Name**
 - i. WQL Query WHERE clause: **Name = "Remote Desktop Session Host"**
4. Click **OK** to create the Global Condition.
5. From the ribbon select Create Global Condition.
6. Fill out the following information:
 - a. Name: **RDSH Installed**
 - b. Device Type: **Windows**
 - c. Condition Type: **Expression**
7. Click **Add Clause...**
8. Fill out the following information:
 - a. Category: **Custom**
 - b. Condition: **RDSH Installed Query**
 - c. Rule Type: **Existential**
 - d. Check the option: **The selected global condition must exist on client devices**
9. Click **OK**
10. Click **OK** to create the Global Condition.

Now a deployment type needs to be created that utilizes the App-V Client for RDS in conjunction with the requirements created above, to deploy only on servers that have the Remote Desktop Session Host role enabled.

1. Navigate to the **Software Library** workspace, select **Applications**.
2. Select the **Microsoft Application Virtualization Client 5** that was imported above.
3. From the ribbon, click **Properties**.
4. Select the **Deployment Types** tab.
5. Select the **Microsoft Application Virtualization (App-V) 5.0 Client Install from Script** deployment type.
6. Click **Copy**.
7. Select the **Microsoft Application Virtualization (App-V) 5.0 Client Install from Script-copy** deployment type.
8. Click **Increase Priority** to move the copied deployment type above the original.
9. Click **Edit**.
10. Rename the deployment type to **Microsoft Application Virtualization (App-V) 5.0 Client Install from Script – RDS**.
11. Select the **Content** tab.
12. Verify the **Content Location** path contains the **appv_client_setup_rds.exe** installer. If it doesn't, click **Browse** and navigate to the correct location.
13. Select the **Programs** tab.
14. Fill in the following information:
 - a. Installation Program: "**appv_client_setup_rds.exe**"
/SHAREDCONTENTSTOREMODE=1 /ENABLEPACKAGESCRIPTS=1 /q /log %temp%\AppV5RDSClientInstall.log
 - b. Uninstall Program: "**appv_client_setup_rds.exe**" **/UNINSTALL /q /norestart /log %temp%\AppV5RDSClientUninstall.log**

Note: The program description above enables shared content store mode and package scripts. All available setup options are available at <http://technet.microsoft.com/en-us/library/jj713460.aspx>.

15. Select the **Requirements** tab.
16. Select the **Operating System** requirement. Click **Edit**.
17. Uncheck all **Windows 7** and **Windows 8** selections. Click **OK**.
18. Click **Add**. Fill out the following information:
19. Fill out the following information:
 - a. Category: **Custom**
 - b. Condition: **RDSH Installed**

- c. Operator: **Equals**
 - d. Value: **True**
20. Click **OK** to add the requirement to the deployment type.
 21. Click **OK**.
 22. Click **OK** to complete the configuration.

This guidance provides a way to deploy the App-V RDS Client with Configuration Manager. There are additional ways to do this, depending on the configuration of the environment, depending on items like Active Directory structure and placement of RDSH Servers. Additional information can be obtained with Global Conditions in Configuration Manager.

Deploying the App-V Client Manually

If Configuration Manager is not being utilized as the deployment method for VDI (session or VM-based) scenarios, administrators can utilize the following information to build an automated script or integrate with their current software deployment solution. VDI (VM based) scenarios require the App-V 5.0 Client for Windows Desktops installation. VDI (session based) scenarios require the App-V 5.0 Client for RDS be installed on the RDHS servers, **only after** the RDSH role has been installed on the server.

Manual install on Windows Desktops

The App-V Client installs on Windows 8 with no additional prerequisites. When installing the App-V Client on Window 7, the following pre-requisites must be installed prior to installing the App-V Client:

- .Net Framework 4 (full)
- KB2533623
- WMF 3.0 (PowerShell 3.0)

On the Windows desktop:

1. Login as an account with local administrative privileges and run **appv_client_setup.exe**.
2. In the **Microsoft Application Virtualization (App-V) Client 5.0 Setup** dialog box, click **Install**.
3. Read and accept the license agreement, and then click **Next**.
4. On the **Microsoft Update Opt In** page, select an option and then click **Next**.
5. On the **Customer Experience Improvement Program** page, select an option and then click **Install**.
6. When the installation is complete, click **Close**.

Note: A number of command line switches are available at time of install for configuring the App-V Client including App-V Publishing server configuration. For the complete list of options, see the Microsoft TechNet: [How to Deploy the Client](#)

Manual install on RDSH Server

The App-V 5.0 Client (desktop client) cannot be installed on Remote Desktop Services Host servers. When deploying to RDSH servers, the **App-V RDS Client** must be installed only after the RDSH role has been installed on the server.

The App-V Client for RDS will install on Windows Server 2012 with no additional prerequisites. When installing the App-V Client for RDS on Window Server 2008 R2, the following pre-requisites must be installed, prior to installing the App-V Client:

- .Net Framework 4 (full)
- KB2533623
- WMF 3.0 (PowerShell 3.0)

Pre-install

As with installing any local application on an RDS host, the host should first be placed into **install** mode by executing the following command from a command prompt:

```
change user /install
```

Install

The following steps should be completed for installation.

1. Launch the executable (**appv_client_setup_rds.exe**).
2. On the **Software License Terms** page, review the license terms. To accept the license terms, select **I accept the license terms**. Click **Next**.
3. On the **Use Microsoft Update to help keep your computer secure and up-to-date** page, to enable Microsoft updates, select **Use Microsoft Update when I check for updates (recommended)**. To disable Microsoft updates, select **I don't want to use Microsoft Update**. Click **Next**.
4. On the **Customer Experience Improvement Program** page, to participate in the program, select **Join the Customer Experience Improvement Program**. This will allow information to be collected about how you use App-V 5.0. If you don't want to participate in the program, select **I don't want to join the program at this time**. Click **Install**.

5. On the **Setup completed successfully** page, click **Close**. After you complete the installation there will be multiple entries in **Programs** for the App-V 5.0 client. There will be an instance for the associated **exe**, **.msi**, and for the **language pack**. Only the **.exe** entry can be uninstalled.

Note: A number of command line switches are available for configuring the App-V Client at time of install, including App-V Publishing server, autoload, and shared content store configurations. For the complete list of options see the Microsoft TechNet: [How to Deploy the Client](#) or refer to the [Deploying the App-V Client via Configuration Manager](#) section.

Post-install

Finish the installation by placing the RDSH server back in execution mode. From the command prompt enter:

```
change user /execute
```

Configuring the App-V Client to use App-V Full Infrastructure

The steps provided below can be automated with setup switches or configured manually, post installation. For the purposes of this appendix, the specific settings are listed as Post-installation tasks to provide details of each individual setting. However, these specific settings can be accomplished with setup switches available from the link above.

App-V Clients utilizing the **App-V Full Infrastructure** for delivery of applications require configuration of the App-V Client to communicate with the appropriate App-V publishing server.

To configure the App-V Client with the App-V Publishing server enter the following commands in an elevated Windows PowerShell console:

```
Import-Module AppvClient  
Add-AppvPublishingServer -URL http://appvserver:889 -Name AppVServer -  
GlobalRefreshEnabled 1 -GlobalRefreshOnLogon 1 -GlobalRefreshInterval 600
```

Note: The *GlobalRefresh* settings are required when deploying virtual applications to machine targets.

Additional App-V Client Configuration Options

Deployment of the App-V Client manually or using Configuration Manager requires knowledge of the specific settings. The settings in this section are additional settings, along with specific

examples of their usage. All of the settings presented can be configured after the installation or as part of the installation with the use of switches for the EXE.

All App-V client-side configurations can now be performed via Windows PowerShell using the **AppvClient** PowerShell module. The PowerShell command **Import-Module AppvClient** must be entered to load the module before entering any App-V Client PowerShell commands.

This table describes commonly used App-V client options (set-appvclientconfiguration) and their specific uses for VDI (session and VM-based) solutions.

Configuration Option	Description	Setting	Recommended Setting
-SharedContentStoreMode	Enables or disables Shared Content Store Mode which disabled the client-side caching of App-V applications	0 (default) Disabled and 1 Enabled Requires a reboot if configured after installation	1 for environments where disk space is limited or expensive (both VDI session and VM-based)
-EnablePackageScripts	Enables the use of scripting in App-V packages	0 (default) disabled and 1 Enabled	1 as most organizations will have at least one application that requires package scripts
-PackageInstallationRoot "drive:\directory"	Defines the local package store location for App-V packages.	%programdata%\App-V (default)	Location on separate drive or partition where additional space is available for package store. When utilizing Shared Content Store Mode this setting has less significance as only targeted applications are cached in the

-AutoLoad	Configures the App-V Client to locally cache virtual applications with 2 which is recommended for use with RDSH clients that are not operating in Shared Content Store Mode.	0 disabled, 1 (default) Previously Launched, 2 All published virtual applications	package store. VDI (session based) solutions without Shared Content Store mode should be set to 2.
------------------	--	---	---

Usage example for a VDI (session based) based server with Shared Content Store Mode as the desired configuration:

```
Set-AppvClientConfiguration -AutoLoad 0 -EnablePackageScripts 1 -
SharedContentStoreMode 1
```

Additional App-V Client configuration options are available that are not directly related to the scope of this document. For more details about these and the remaining configuration options, see the help documentation by entering the following commands at the PowerShell prompt:

```
Import-Module AppvClient
Get-Help Set-AppvClientConfiguration -detailed
```

Note: Many more configuration options are available. For a complete list of App-V Client options that can be configured using Windows PowerShell see the Microsoft TechNet article: [About Client Configuration Settings](#)